

Requested Patent: JP3070032A

Title: EXTERNAL NAME INFORMATION PROCESSING SYSTEM IN LINKING ;

Abstracted Patent: JP3070032 ;

Publication Date: 1991-03-26 ;

Inventor(s): GOTO ATSUSHI ;

Applicant(s): NEC CORP ;

Application Number: JP19890205415 19890808 ;

Priority Number(s): ;

IPC Classification: G06F9/06 ;

Equivalents: ;

ABSTRACT:

PURPOSE: To shorten link processing time by moving the position of a hash entry on a synonym chain corresponding to an external name to a position nearer to a header when the external name exists on the synonym chain in reference to the external name.

CONSTITUTION: When the external names of object modules 101-103 inputted in linking are referred by using a hash table 15, the position of the hash entry corresponding to the external name on the synonym chain is moved to the position nearer to the header if a corresponding external name exists on the synonym chain instructed by the hash table 15. Therefore, the position of the external name on the synonym chain inputted afterwards is carried up to the position nearer to the header in each every case when reference is performed frequently. Thereby, the wasteful frequencies of comparison for the external name in the retrieval of the synonym chain can be reduced, which accelerates a link processing.

SVL920030040454

⑫ 公開特許公報(A) 平3-70032

⑤ Int. Cl.³

G 06 F 9/06

識別記号

4 1 0 E

庁内整理番号

7361-5B

⑬ 公開 平成3年(1991)3月26日

審査請求 未請求 請求項の数 1 (全4頁)

⑭ 発明の名称 リンク時外部名情報処理方式

⑮ 特 願 平1-205415

⑯ 出 願 平1(1989)8月8日

⑰ 発 明 者 後 藤 淳 東京都港区芝5丁目33番1号 日本電気株式会社内

⑱ 出 願 人 日本電気株式会社 東京都港区芝5丁目7番1号

⑲ 代 理 人 弁理士 井ノ口 壽

明 細 書

1. 発明の名称

リンク時外部名情報処理方式

2. 特許請求の範囲

リンク時に入力したオブジェクトモジュール内の外部名を参照するためのハッシュテーブルと、前記ハッシュテーブルを利用して前記外部名を参照する際に、該当する外部名が前記ハッシュテーブルから指示されるシノニムチェーン上にあるか否かを判定するためのハッシング処理手段と、前記該当外部名が前記シノニムチェーン上にあった場合、前記該当外部名に対応するハッシュエントリのシノニムチェーン上の位置をよりヘッダに近い位置に移動させるためのリンク処理手段とを具備して構成したことを特徴とするリンク時外部名情報処理方式。

8. 発明の詳細な説明

(産業上の利用分野)

本発明は計算機システムにおいて複数のオブ

ジェクトモジュールから一個のロードモジュールを生成する方式に関し、特にその際のリンク方式に関する。

(従来技術)

従来、リンク処理において入力されたオブジェクトモジュールに含まれる定義、あるいは参照等の外部名に関し、定義と参照とを関連付けを行ったり、あるいは新たに入力されたオブジェクトモジュールに含まれる定義名がすでに他にオブジェクトモジュール内の定義によつて使用済みであるか否かの検査を行ったりする場合、通常、ハッシング手法が用いられている。

ハッシングとは、名前を文字コードの列とし、それに対してハッシュ関数と呼ばれる処理を行うことにより、ハッシュ値と呼ばれる数値を算出し、その数値をもとにして名前を管理する手法である。

ハッシングにおいて、異なる名前でも同一のハッシュ値を持つ名前同志をシノニムと称する。通常、シノニム同士は、ハッシュテーブル上に

チェーンヘッダを持つシノニムチェーンと呼ばれる一本のチェーンに繋がれるのが普通である。

目的とするハッシュエントリを得るためには、まず外部名からハッシュ関数によつてハッシュ値を算出し、それによつてハッシュテーブルからシノニムチェーンのヘッダを求める。次に、シノニムチェーンを順に進めてゆき、目的とするハッシュエントリを検索する。

また、新たに外部名を追加する場合には、シノニムチェーンの最後に追加するのが普通である。

(発明が解決しようとする課題)

上述した従来のリンク処理における外部名のハッシング処理においては、シノニムチェーン上のハッシュエントリ内の外部名と、目的とする外部名とが一致するまで順次繰り返して比較することによりハッシュエントリの検索を行っている。

従つて、シノニムチェーン上の後方にあるハッシュエントリほど無駄な比較が増大し、検索

に時間がかかることになると云う欠点がある。

また、従来シノニムチェーン上に新たなハッシュエントリを追加する場合には、シノニムチェーンの最後に追加するのが普通であつた。このため、後に入力されたオブジェクトモジュール内の外部名ほどシノニムチェーンの後方に繋がれる可能性が増大し、その結果、検索に時間がかかる可能性が増加することになると云う欠点がある。

このような場合に、たまたま遅れて入力されたためにシノニムチェーンの後方に繋がれた外部名に対して、以後、参照が多発した場合には、ハッシュエントリの検索に必要とされる時間が非常に増大することになると云う欠点がある。

以上のように、従来の方式においては外部名の入力順序によつてはシノニムチェーンの検索に要する時間が増大し、それによつてリンク処理に要する時間が増大するという欠点があつた。このため、それを解決するという課題がある。

本発明の目的は、リンク時に入力したオブジ

ェクトモジュール内の外部名をハッシュテーブルを利用して参照する際に、該当する外部名がハッシュテーブルから指示されるシノニムチェーン上にあつた場合、その外部名に対するハッシュエントリのシノニムチェーン上の位置をよりヘッダに近い位置に移動させることにより上記欠点を除去し、リンク処理時間を短縮できるように構成したリンク時外部名情報処理方式を提供することにある。

(課題を解決するための手段)

本発明によるリンク時外部名情報処理方式はハッシュテーブルと、ハッシング処理手段と、リンク処理手段とを具備して構成されている。

ハッシュテーブルは、リンク時に入力したオブジェクトモジュール内の外部名を参照するためのものである。

ハッシング処理手段は、ハッシングテーブルを利用して外部名を参照する際に、該当する外部名がハッシュテーブルから指示されるシノニムチェーン上にあるか否かを判定するためのも

のである。

リンク処理手段は、該当外部名がシノニムチェーン上にあつた場合、該当外部名に対応するハッシュエントリのシノニムチェーン上の位置をよりヘッダに近い位置へ移動させるためのものである。

(実施例)

次に、本発明について図面を参照して説明する。

第1図は、本発明によるリンク時外部名情報処理方式の一実施例を示すブロック図である。

第1図において、1はリンク処理方式、11はオブジェクトモジュール入力処理手段、12はリンク処理手段、13はハッシング処理手段、14はロードモジュール生成処理手段、15はハッシュテーブル、101~103はそれぞれ第1、第2、および第nのオブジェクトモジュール、104はロードモジュールである。

リンク処理方式1はオブジェクトモジュール1~nをリンクし、ロードモジュールLMを生

成する処理方式である。

まず、オブジェクトモジュール入力処理手段11で第1～第nのオブジェクトモジュール101～103を入力する。

次に、リンク処理手段12で外部名を処理するが、この際、ハッシングを行うためにハッシング処理手段13を呼出す。

第2図は、ハッシング処理手段13の動作概要を示したフローチャートであり、第3図はハッシュテーブル15の周辺の構造を示した説明図である。

第2図において、まずステップH1で処理対象が定義であるか、あるいは参照であるかの判定を行い、定義の場合にはそれぞれステップH2、H3でハッシュ値の算出、およびシノニムチェーンの検索を行う。

次に、ステップH4においてシノニムチェーン上に目的とする外部名を持つハッシュエントリが存在したか否かの判定を行い、存在しなかった場合にはステップH5、H6において新た

H13、H14において新たにハッシュエントリを生成し、対応するシノニムチェーンの最後に追加する。

シノニムチェーン上に目的とする外部名を持つハッシュエントリが存在した場合には、ステップH16において、そのハッシュエントリのシノニムチェーン上の位置を、よりシノニムチェーンヘッダに近い位置にひとつ分だけ繰上げる。

最後に、ステップH15においてハッシュエントリに参照情報をセットするが、参照情報は定義情報と異なり、複数存在する可能性がある。このため、参照情報チェーンの最後に参照情報を追加するようにする。

以上で、ハッシング処理手段13の動作を終了する。

第3図において、ハッシュテーブルはシノニムチェーンヘッダの集合であり、それをチェーンヘッダとしてシノニムチェーンが構成される。

シノニムチェーンは、ハッシュエントリのチ

ンにハッシュエントリを生成して、対応するシノニムチェーンの最後に追加する。

もし、シノニムチェーン上に目的とする外部名を持つハッシュエントリが存在した場合には、その外部名に対して定義情報が既に存在するか否かを判定し、もしあれば多重定義としてエラーとする。

シノニムチェーン上に目的とする外部名を持つハッシュエントリが存在しないか、あるいは存在しても定義情報がなかった場合には、この外部名を持つ定義が初めて現れたことになる。このため、ハッシュエントリ上に定義情報をセットして定義側の処理を終了する。

処理対象が参照の場合には、ステップH10、H11でハッシュ値の算出、およびシノニムチェーンの検索を行う。

次に、ステップH12においてシノニムチェーン上に目的とする外部名を持つハッシュエントリが存在したか否かの判定を行い、ハッシュエントリが存在しなかった場合には、ステップ

エンであり、各ハッシュエントリには対応する外部名情報/定義情報を指すポインタ、および参照情報チェーンのヘッダを含んでいる。

また、定義情報と参照情報とは、どちらか一方のみしか存在しない場合がある。

最後に、ロードモジュール生成処理手段14においてロードモジュールLMを生成して、リンク処理手段1の動作を終了する。

(発明の効果)

以上説明したように本発明は、リンク時に入力したオブジェクトモジュール内の外部名をハッシュテーブルを利用して参照する際に、該当する外部名がハッシュテーブルから指されるシノニムチェーン上にあつた場合、その外部名に対するハッシュエントリのシノニムチェーン上の位置をよりヘッダに近い位置に移動させることにより、後から入力された外部名に対しても参照が頻繁に発生する場合には、その都度、シノニムチェーン上の位置がシノニムチェーンヘッダに近い位置に繰上がって行く。このため、

シノニムチェーンを検索する際の無駄な外部名の比較回数が減少し、リンク処理を高速化することが可能になるという効果がある。

4. 図面の簡単な説明

第1図は、本発明によるリンク時外部名情報処理方式の一実施例を示すブロック構成図である。

第2図は、ハッシング処理の概要を示したフローチャートである。

第3図は、ハッシュテーブル周辺の構造を示した説明図である。

1 … リンク処理方式

11 … オブジェクトモジュール入力処理手段

12 … リンク処理手段

13 … ハッシング処理手段

14 … ロードモジュール生成処理手段

15 … ハッシュテーブル

101 ~ 103 … オブジェクトモジュール

104 … ロードモジュール

H1 ~ H16 … 処理ステップ

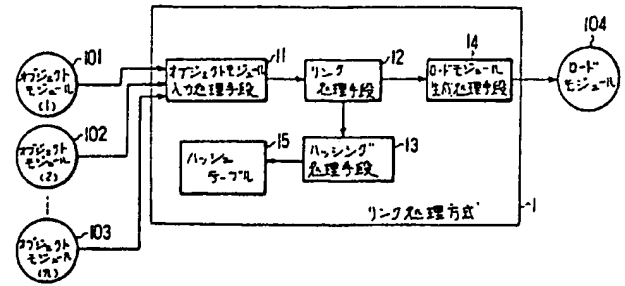


図 3

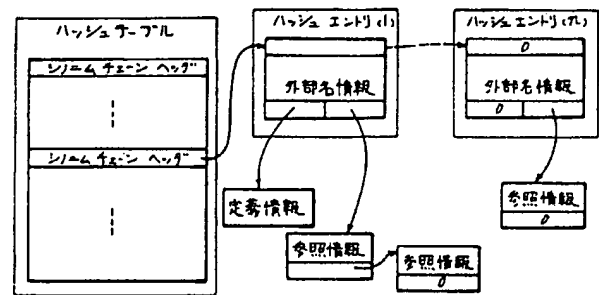


図 2

